



# Latency-Based Probabilistic Information Processing in Recurrent Neural Hierarchies

Alexander Gepperth, Mathieu Lefort

## ► To cite this version:

Alexander Gepperth, Mathieu Lefort. Latency-Based Probabilistic Information Processing in Recurrent Neural Hierarchies. International Conference on Artificial Neural Networks (ICANN), Sep 2014, Hamburg, Germany. pp.715 - 722, 10.1007/978-3-319-11179-7\_90 . hal-01098699

**HAL Id: hal-01098699**

**<https://hal.inria.fr/hal-01098699>**

Submitted on 28 Dec 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Latency-based probabilistic information processing in recurrent neural hierarchies

Alexander Gepperth, Mathieu Lefort

ENSTA ParisTech, 828 Blvd des Marechaux, 91762 Palaiseau, France (email: [firstname.lastname@ensta-paristech.fr](mailto:firstname.lastname@ensta-paristech.fr)) and INRIA FLOWERS, 100 avenue de la Liberation, 44801 Talence, France.

**Abstract.** In this article, we present an original neural space/latency code, integrated in a multi-layered neural hierarchy, that offers a new perspective on probabilistic inference operations. Our work is based on the dynamic neural field paradigm that leads to the emergence of activity bumps, based on recurrent lateral interactions, thus providing a spatial coding of information. We propose that lateral connections represent a data model, i.e., the conditional probability of a “true” stimulus given a noisy input. We propose furthermore that the resulting attractor state encodes the most likely “true” stimulus given the data model, and that its latency expresses the confidence in this interpretation. Thus, the main feature of this network is its ability to represent, transmit and integrate probabilistic information at multiple levels so that to take near-optimal decisions when inputs are contradictory, noisy or missing. We illustrate these properties on a three-layered neural hierarchy receiving inputs from a simplified robotic object recognition task. We also compare the network dynamics to an explicit probabilistic model of the task, to verify that it indeed reproduces all relevant properties of probabilistic processing.

## 1 Introduction

With the advent of Bayesian inference accounts of biological information processing [1], a large body of literature [2–6] focuses on probabilistic aspects of neural coding. Most authors explicitly assume that neural population activity is related to probability distributions. Such an assumption faces two major problems trying to give account of neural processing. First, such a coding, which needs multiplication of probabilistic values, collides with summation based neural computation. To overcome this conflict, a very influential idea posits that single-neuron activity is related to log-probability [4, 5], which would allow to perform multiplications by summation. An alternative approach [3] is to consider single neuron’s firing rates as the realizations of Poisson-like random variables. Under certain conditions, sums of such variables come from a distribution whose mean corresponds to the product of individual means, thus realizing a multiplication by summation. Second, probabilistic approaches generally work on the single-neuron level without reference to other neurons in the same population [2–5]. However, it is a fact that biological processing makes heavy use of lateral connections and population coding.

We aim to overcome these limitations by taking another way: we propose to process probabilistic information in population based processing using the latency of attractor states. There is converging evidence from both physiological [7–10] and behavioral [11, 12] investigations that latency plays a role in the neural encoding of information. Neurons in the striate cortex, for example, encode stimulus contrast into response latency [8], whereas decision making processes typically take longer depending on the number of conflicting alternatives [11], conceivably reflecting increased response latency on the neural level. Similar effects have also been observed in language processing [12].

Our work is based on the dynamic neural field paradigm [13–15] of recurrent interactions through lateral connections. We posit that lateral connections implement a data model expressing the conditional probability  $p(\mathbf{M}|\mathbf{S})$  of an underlying "true" stimulus  $\mathbf{M}$  given the noisy/mixed/corrupted input  $\mathbf{S}$ . The competitive neural field dynamics will converge to an attractor state maximally compatible with the input and the data model, see [16]. Furthermore, it is a well-documented effect [17, 13] that latency of this process varies depending on inputs, and we posit that it encodes the match of input to data model. In this combined space/latency code [18], the position of the emergent activity represents the most likely "true" stimulus  $\mathbf{M}^* = \arg \max p(\mathbf{M}|\mathbf{S})$  and the latency expresses its probability under the data model,  $p(\mathbf{M}^*|\mathbf{S})$ , which we term "confidence". Thus, neural populations are not viewed as representing full probability distributions as sub-leading interpretations are suppressed by competition.

While in a previous publication [18] we demonstrated that such our combined space/latency neural code can implement Bayes-optimal decision making in a simple setting, this article is dedicated to the study of this coding in a deeper hierarchy with more complex inputs inspired by works on a simple robotic object recognition task [17]. In particular, we wish to analyze the effects of model violations (which we will often denote by "uncertainty"), focusing on the encoding of uncertainty into response latency, the transport and accumulation of latency across hierarchy levels, and the decoding of latency into optimal local decisions. Analogies to an exact probabilistic model will be discussed as well.

## 2 Model

### 2.1 Architecture

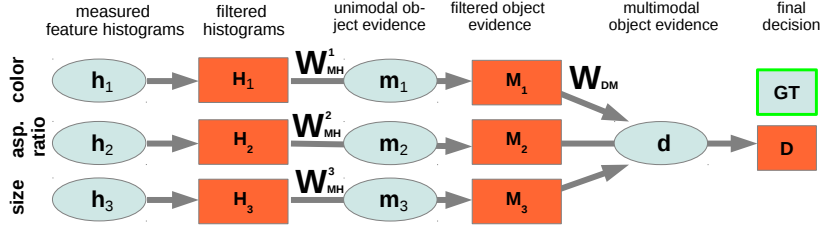
We use a three layered multimodal architecture (see figure 1) as a support of our study on our combined space/latency neural code in hierarchical architecture. Feed-forward connection between maps are plastic and each map of the hierarchy evolves according to a very general mechanisms of competitive neural dynamics so that our study should be valid across a wide range of spiking and non-spiking neural models.

### 2.2 Model Equations

*Activity computation* The field activity  $L$  ( $L \in \{H_1, H_2, H_3, M_1, M_2, M_3, D\}$ ) at position  $x$  and time  $t$  is equal to  $f[u^L(x, t)]$  with the quantity  $u^L$  that represents

## Latency-based probabilistic processing in recurrent neural hierarchies

3



**Fig. 1.** Neural hierarchy which is the basis for all investigations in this article. Each orange box corresponds a neural field and each oval represents the corresponding neural field input which is basically a weighted sum of the neural activity of the previous layer or input from the environment (see section 2.2 for details).  $GT$  is the ground truth that describes the current object corresponding to input features  $h_1, h_2$  and  $h_3$ .

the membrane potential of the field which evolves according to a slightly modified version of dynamic neural field proposed in [14]:

$$\tau \dot{u}^L(x, t) = -u^L(x, t) + \alpha g^L[S^L(x, t)] + \beta (w(x - x') * f[u(x', t)]) + \gamma \sigma(x, t) + h$$

with  $w$  the fixed lateral interaction kernel,  $f[u]$  the point-wise applied transfer function given by  $f[u] = \left(1 + \exp\left(\frac{-2(u-\theta)}{\nu}\right)\right)^{-1}$ ,  $\sigma$  normally distributed white noise,  $h$  the resting potential and  $\tau$  the time scale of field evolution.  $S^L$  is the afferent input:

$$S^L = \begin{cases} \mathbf{h}_i & \text{if } L = H_i \\ \mathbf{m}_i = \sigma(\mathbf{W}_{MH}^i \cdot \mathbf{u}^{H_i}) & \text{if } L = M_i \\ \mathbf{d} = \sigma(\sum_i \mathbf{W}_{DM}^i \cdot \mathbf{u}^{M_i}) & \text{if } L = D \end{cases}$$

with  $\sigma$  a logistic transfer function (refer to figure 1 for other notations). We use an input transfer function  $g^L[S]$  so that inputs can excite the field effectively:

$$g^L[S^L] = \begin{cases} k^L S^L & \text{if } k^L S^L \leq 1 \\ 1 & \text{else} \end{cases} \quad (1)$$

with a suitably chosen constant  $k^L$  that depends on average input strength. The coefficients  $\alpha, \beta$  and  $\gamma$  respectively determine the contribution of afferent input, lateral recurrent interactions and noise. The interaction kernel  $w$  is usually chosen to be symmetric:  $w(dx) = a_0 G_{\mu=0, \sigma=\sigma_{\text{on}}}(dx) - b_0 G_{\mu=0, \sigma=\sigma_{\text{off}}}(dx) - c_0$ , where  $G_{\mu, \sigma}$  denotes a Gaussian with mean  $\mu$  and standard deviation  $\sigma$ , and  $\sigma_{\text{on}} < \sigma_{\text{off}}$ . The constants  $a_0, b_0, c_0$  are chosen suitably to achieve the desired level of local excitation/inhibition ( $a_0, b_0$ ) as well as global inhibition ( $c_0$ ). To ensure numerical stability, we clip the neural field potentials  $u$  whenever they exceed the range defined by  $[u_{\text{min}}, u_{\text{max}}]$ . Please notice that all model parameters are identical for each field of the model except for  $g^L$  function that needs to be tuned for each field so that input is sufficiently high to trigger an activity.

4 A.Gepperth, M.Lefort

*Weights learning* All feed-forward weights between fields are trained by online logistic regression [19] with logistic transfer function  $\sigma$ , using a step size  $\lambda_{LR}$ , where the learning target is always the population-encoded object identity GT.

*Probabilistic model* Using the notation of Fig. 1, we introduce the following shorthand:  $\mathfrak{M} \equiv \{\mathbf{M}_i\}$ ,  $\mathfrak{m} \equiv \{\mathbf{m}_i\}$ ,  $\mathfrak{H} \equiv \{\mathbf{H}_i\}$  and  $\mathfrak{h} \equiv \{\mathbf{h}_i\}$ . The probability for the presence of a certain object as a function of the inputs is thus expressed as  $p(\mathbf{D}|\mathfrak{h})$ . Using the law of total probability, we can introduce the other network quantities into this expression:

$$p(\mathbf{D}|\mathfrak{h}) = \sum_{\{\mathbf{d}_j\}} \sum_{\{\mathfrak{M}_k\}} \sum_{\{\mathfrak{m}_l\}} \sum_{\{\mathfrak{H}_m\}} p(\mathbf{D}|\mathbf{d}_j) p(\mathbf{d}_j|\mathfrak{M}_k) p(\mathfrak{M}_k|\mathfrak{m}_l) p(\mathfrak{m}_l|\mathfrak{H}_m) p(\mathfrak{H}_m|\mathfrak{h}) \quad (2)$$

where the sums run over all possible realizations of each variable. Since the transformations  $\mathfrak{M} \rightarrow \mathbf{d}$  and  $\mathfrak{H} \rightarrow \mathfrak{m}$  are deterministically governed by the network weights  $W_{MH}^i$  and  $W_{DM}^i$  (see Fig. 1), the corresponding distributions  $p(\mathbf{d}|\mathfrak{M})$  and  $p(\mathfrak{m}|\mathfrak{H})$  are delta-like and the sums over these variable thus vanish. Due to the splitting of the hierarchy into independent modal flows (see Fig. 1), the expressions  $p(\mathfrak{H}_m|\mathfrak{h})$  and  $p(\mathfrak{M}_k|W_{MH}\mathfrak{H})$  factorize into products of the unimodal probabilities. In order to better approximate the network dynamics, we suppose that the sums over conditional probabilities are approximated by their maxima  $\mathfrak{M}_k^* \equiv \arg \max p(\mathfrak{M}_k|W_{MH}\mathfrak{H})$  and  $\mathfrak{H}_k^* \equiv \arg \max p(\mathfrak{H}_k|\mathfrak{h})$ . Thus:

$$\begin{aligned} p(\mathbf{D}|\mathfrak{h}) &= \sum_{\{\mathfrak{M}_k\}} \sum_{\{\mathfrak{H}_m\}} p(\mathbf{D}|\mathbf{W}_{DM}\mathfrak{M}_k) p(\mathfrak{M}_k|W_{MH}\mathfrak{H}_m) p(\mathfrak{H}_m|\mathfrak{h}) \approx \\ &\approx p(\mathbf{D}|\mathbf{W}_{DM}\mathfrak{M}^*) p(\mathfrak{M}^*|W_{MH}\mathfrak{H}^*) p(\mathfrak{H}^*|\mathfrak{h}) \equiv \\ &\equiv \text{conf}(\mathbf{D}) \Pi_i \text{conf}(\mathbf{M}_i) \Pi_j \text{conf}(\mathbf{H}_j). \end{aligned} \quad (3)$$

As each of the terms in Eqn. (3) represents a confidence tied to a specific neural field, it is very natural to associate them with response latencies. Please note that, in the highest layer  $\mathbf{D}$ , decisions and confidences are effectively dissociated as the same decision can be reached for many confidence combinations. Each confidence contributes equally to the final expression, independently of its hierarchical position. The functional form of confidences will depend generally on the problem, in our case it would punish deviations from a unimodal distribution.

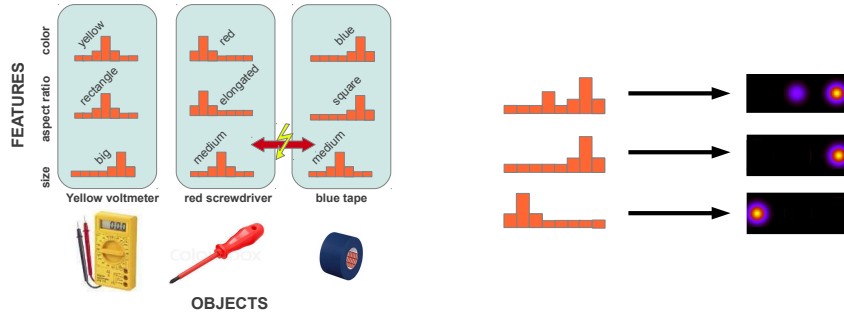
### 3 Experiments

#### 3.1 Temporal organization of a single input presentation

Feature histograms  $\mathbf{h}_i$  are presented to the lowest hierarchy levels  $H_i$  of the network at time  $t_0 + 1$  and maintained for a total of  $T$  simulation steps. Directly before this happens, at time  $t_0$ , all field potentials in the hierarchy are reset to the resting potential  $h$ , see Sec. 2.2. Subsequently, field potentials and weights evolve freely according to the dynamic models described in Sec. 2.2

### 3.2 Input stimuli

We simulate a robotic object recognition task containing the objects "red screwdriver", "yellow voltmeter" and "blue tape" (see Fig. 2). We generate synthetic "measurements" in the modalities of color, aspect ratio and size for a random succession of these three objects drawn from a uniform distribution. Following[17], measurements are modeled as feature histograms over different modalities. These histograms contain a single Gaussian peak of amplitude  $A_S = 1.0$ , at a position determined by object identity as shown in Fig. 2. To serve as inputs, these one-dimensional histograms are encoded along one axis of a two-dimensional image of dimensions  $(60, 10)^T$  using the convolution coding technique with a Gaussian basis function of standard deviation of 3 pixels, see Fig. 2(right). The layers



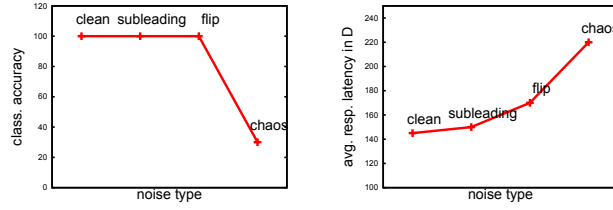
**Fig. 2.** Simplified synthetic object recognition task. Left: objects and their visual properties (expressed as feature histograms) in the three modalities color, aspect ratio and size. Please note the overlap in the "size" modality between screwdriver and tape, leading to potential ambiguities in feedforward processing. Right: population-encoding of these one-dimensional histograms input.

$H_i$ ,  $M_i$  and  $D$  are connected as indicated in Fig. 1. The length of a single input presentation is set to  $T = 200$ , the learning rate for all connections is  $\lambda_{LR} = \frac{0.05}{60 \cdot 100}$ . New inputs, as shown in Fig.2 arrive every  $T$  iterations, draw from a uniform object distribution. In the beginning there is a learning phase of 12000 iterations corresponding to 20 presentations per object, after which learning is disabled for performing experiments. We choose a uniform parametrization of neural field layers of size 60x10 (see Sec. 2.2):  $\tau = 15$ ,  $\theta = 0$ ,  $\nu = 2.5$ ,  $\alpha = 1$ ,  $\beta = 4$ ,  $\gamma = 0.11$ ,  $\sigma_{on}=3$ ,  $\sigma_{off} = 6$ ,  $a_0 = b_0 = 1$ ,  $c_0 = 0.55$ ,  $h = -1$ . Zero-padding boundary conditions are used for all lateral interactions. The input transfer function constant  $k_I$  is set to 1 for the fields  $H_i$ , to 1.8 for the fields  $M_i$  and to 1.3 for the field  $D$ . Response latency is defined as the time until an activity  $\geq 0.9$  is observed.

### 3.3 Classification performance

We measure the ability of the network to recognize the currently "presented" object when various levels of noise are applied, which can be of the types "clean"

6 A.Gepperth, M.Lefort

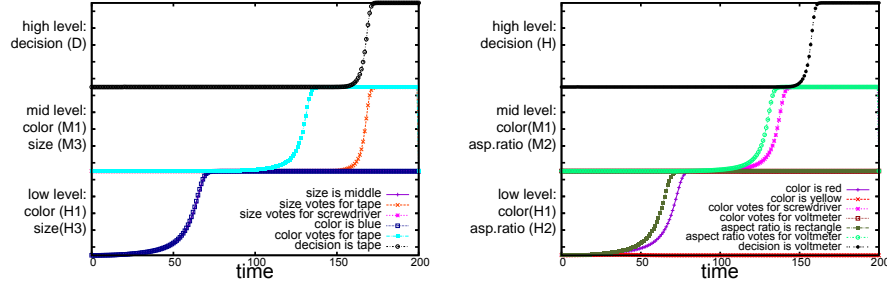


**Fig. 3.** Development of classification accuracy (left) and average response latency (right) as a function of noise levels.

(no noise), "sub-leading noise" (small peak of strength 0.5 introduced at a random position in all  $h_i$ ), "flip" (peak in the "color" modality is flipped to a wrong position) and "chaos" (peaks in the color and aspect ratio modalities are randomly switched to a wrong position). Fig .3 shows the corresponding classification rates and average response latencies for these four cases. It is clearly visible that latencies increase in the "sub-leading noise" case and even more for the case "flip", even though classification accuracies stay at 100%. This very nicely reflects the probabilistic nature of processing, as the network can signal something was wrong, and to what degree, even though it delivers perfect performance. For the "chaos" case, evidently no above-chance classification is possible as two-thirds of the modalities are corrupted, which is again reflected in the strongly increased latencies for this case.

### 3.4 Case studies

Using the "clean" condition described in the previous section, we investigate the reaction of the hierarchy to an inherent ambiguity which stems from the fact the size "medium" votes for both "blue tape" and "red screwdriver" in the mid-level of the hierarchy. Instead of one peak of amplitude 1.0, there will now be two peaks of amplitude 0.5 in  $m_3$ . As seen in Fig. 4, activity will still appear in  $M_3$  since the input transfer function of that field scales inputs to a sufficient strength. However, as a consequence of competition, the response will be delayed, reflecting its lower confidence (i.e., probability under the data model). This in turn will delay activity buildup in the highest layer  $D$ , expressing that the top-level decision is not as certain as it could be. Indeed, the vote of  $M_3$  is not really taken into account in defining the response of  $D$  as it comes too late, demonstrating the basic principle of probabilistic information processing in this architecture: later-coming inputs have less influence in attractor formation in recurrent layers. Going beyond the "sub-lading noise" and "flip" conditions, we now investigate what happens when the feature histogram in a single modality is ambiguous and also incorrect: for presentations of the "yellow voltmeter" object, we put a Gaussian of strength 1.0 at the (incorrect) position "red", as well as a Gaussian of strength 0.8 at the (correct) position "yellow" in the color histogram input  $h_1$ , simulating a measurement ambiguity leading to a locally



**Fig. 4.** Selected demonstrations of latency effects. Graphs are organized in three rows corresponding the temporal development of activity in different hierarchy levels. Left: reaction to the ambiguous "blue tape" object. The unambiguous fields  $H_1$  (color) and  $H_3$  (size) have exactly the same latency. Middle row: unimodal object field  $M_1$  (color) and  $M_3$  (size), the latter having a strongly increased latency due to the ambiguity in object definitions. The lacking input strongly delays activity buildup in the top-level decision field  $D$ , which nevertheless occurs since the other modalities are unambiguous. Right: presenting the "yellow voltmeter" object with corrupted histogram input in the color modality. Activity in layers  $H_1$  (color) and  $H_2$  (aspect ratio) differs in latency as color is corrupted. Activity in unimodal object layers  $M_1$  (color) and  $M_2$  (aspect ratio) retains the low-level latency difference as no ambiguity is present at this level. Top row: final decision of the network, expressed by delayed activity in layer  $D$ .

wrong conclusion, see Fig. 4(right). This wrong conclusion is propagated forward to  $M_1$  where it activates the "screwdriver" population. As the other modalities  $M_i, i \neq 1$  vote for the correct object (yellow voltmeter), a correct high-level decision will still be taken in  $D$ .

## 4 Discussion and Outlook

*Comparison to probabilistic model* This article is based on a novel probabilistic interpretation of neural activities making use of biologically plausible neural dynamics. As it is the case with the probabilistic model of eqn.(3), model violations/uncertainty may, with equal influence, originate at any place in the hierarchy as observed in Fig. 4 where uncertainty arising on the lowest or middle layers is transported to the top. This also shows that that confidence from any layer is propagated to the top without change if other layers do not add to it, which is another corollary of eqn.(3). We furthermore observe that not only the fact but also the strength of model violations is transported to the top-level regardless of their origin, although the mapping from conditional probability values to latency is not linear. In addition, the development of response latencies depending on noise, see Sec. 3.3, directly show that decisions and confidences are dissociated as in the model of eqn.(3), thus forming two independent coding dimensions. Lastly, it may be stated that the approximation (and thus the



8 A.Gepperth, M.Lefort

information loss) accepted in eqn.(3), as well as in neural field dynamics, is one that will not usually have negative effects unless a significant part of hierarchy inputs is corrupted.

*Outlook* We intend to limit information loss in our model by "re-activating" suppressed inputs by feedback when high-level decisions are consistent with what was suppressed. Another focus will be the application of this model to more difficult and realistic tasks, the automatic tuning of the various coupling constants by homeostatic processes, and the learning of data models.

## References

1. AL Yuille and HH Bulthoff. Bayesian decision theory and psychophysics. In *Perception as Bayesian Inference*, pages 123–161. University Press, 1996.
2. R. S. Zemel, P. Dayan, and A. Pouget. Probabilistic interpretation of population codes. *Neural Comput*, 10(2):403–430, Feb 1998.
3. WJ Ma, J Beck, P Latham, and A Pouget. Bayesian inference with probabilistic population codes. *Nature Neuroscience*, 9(11), 2006.
4. Rajesh P N Rao. Bayesian computation in recurrent neural circuits. *Neural Comput*, 16(1):1–38, Jan 2004.
5. J. Gold and M. Shadlen. Neural computations that underlie decisions about sensory stimuli. *Trends Cogn Sci*, 5(1):10–16, Jan 2001.
6. David C Knill and Alexandre Pouget. The bayesian brain: the role of uncertainty in neural coding and computation. *Trends Neurosci*, 27(12):712–719, Dec 2004.
7. M. W. Oram, D. Xiao, B. Drietschel, and K. R. Payne. The temporal resolution of neural codes: does response latency have a unique role? *Philos Trans R Soc Lond B Biol Sci*, 357(1424):987–1001, Aug 2002.
8. D. S. Reich, F. Mechler, and J. D. Victor. Temporal coding of contrast in primary visual cortex: when, what, and why. *J Neurophysiol*, 85(3):1039–1050, Mar 2001.
9. Roozbeh Kiani, Hossein Esteky, and Keiji Tanaka. Differences in onset latency of macaque inferotemporal neural responses to primate and non-primate faces. *J Neurophysiol*, 94(2):1587–1596, Aug 2005.
10. T Michelet, GH Duncan, and P Cisek. Response competition in the primary motor cortex: Corticospinal excitability reflects response replacement during simple decisions. *Journal of Neurophysiology*, 104(1), 2010.
11. E Hazeltine, RA Poldrack, and JDE Gabrieli. Neural activation during response competition. *Journal of Cognitive Neuroscience*, pages 118–129, 2000.
12. R. Borowsky and M.E.J. Masson. Semantic ambiguity effects in word identification. *J. Exp. Psych: Learning, Memory, and Cognition*, 22(1):63, 1996.
13. W. Erlhagen and G. Schöner. Dynamic field theory of movement preparation. *Psychological review*, 109(3):545, 2002.
14. Shun-ichi Amari. Mathematical foundations of neurocomputing. *Proceedings of the IEEE*, 78(9):1441–1463, 1990.
15. Paul Cisek. Integrated neural processes for defining potential actions and deciding between them: a computational model. *J Neurosci*, 26(38):9761–9770, Sep 2006.
16. Shigeru Kubota and Kazuyuki Aihara. Analyzing global dynamics of a neural field model. *Neural Processing Letters*, 21, 2005.
17. C Faubel and G Schöner. Learning to recognize objects on the fly: a neurally based dynamic field approach. *Neural Networks*, 21(4):Pages 562–576, 2008.
18. A Gepperth. Processing and transmission of confidence in recurrent neural hierarchies. *Neural Processing Letters*, 2013.
19. CM Bishop. *Pattern recognition and machine learning*. Springer New York, 2006.